$ifclassloadedbeamer$

Gradient Descent

Nipun Batra

IIT Gandhinagar

July 29, 2025

Gradient denotes the direction of steepest ascent or the direction in which there is a maximum increase in f(x,y)

Gradient denotes the direction of steepest ascent or the direction in which there is a maximum increase in f(x,y)

$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} = \begin{bmatrix} 2x \\ 2y \end{bmatrix}$$

Stochastic Gradient Descent

- In SGD, we update parameters after seeing each each point
- Noisier curve for iteration vs cost

Stochastic Gradient Descent

- In SGD, we update parameters after seeing each each point
- Noisier curve for iteration vs cost
- For a single update, it computes the gradient over one example. Hence lesser time

For t iterations, what is the computational complexity of our gradient descent solution?

For t iterations, what is the computational complexity of our gradient descent solution?
Hint, rewrite the above as: $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top \mathbf{X} \boldsymbol{\theta} + \alpha \mathbf{X}^\top \mathbf{y}$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top \mathbf{X} \boldsymbol{\theta} + \alpha \mathbf{X}^\top \mathbf{y}$

Complexity of computing $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(dn)$

For t iterations, what is the computational complexity of our gradient descent solution?
Hint, rewrite the above as: $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top \mathbf{X} \boldsymbol{\theta} + \alpha \mathbf{X}^\top \mathbf{y}$
Complexity of computing $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(dn)$
Complexity of computing $\alpha \mathbf{X}^\top \mathbf{y}$ once we have $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(d)$ since $\mathbf{X}^\top \mathbf{y}$ has d entries

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top \mathbf{X} \boldsymbol{\theta} + \alpha \mathbf{X}^\top \mathbf{y}$

Complexity of computing $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(dn)$

Complexity of computing $\alpha \mathbf{X}^\top \mathbf{y}$ once we have $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(d)$ since $\mathbf{X}^\top \mathbf{y}$ has d entries

Complexity of computing $\mathbf{X}^\top \mathbf{X}$ is $\mathcal{O}(d^2 n)$ and then multiplying with α is $\mathcal{O}(d^2)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top \mathbf{X} \boldsymbol{\theta} + \alpha \mathbf{X}^\top \mathbf{y}$

Complexity of computing $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(dn)$

Complexity of computing $\alpha \mathbf{X}^\top \mathbf{y}$ once we have $\mathbf{X}^\top \mathbf{y}$ is $\mathcal{O}(d)$ since $\mathbf{X}^\top \mathbf{y}$ has d entries

Complexity of computing $\mathbf{X}^\top \mathbf{X}$ is $\mathcal{O}(d^2 n)$ and then multiplying with α is $\mathcal{O}(d^2)$

All of the above need only be calculated once!

For each of the t iterations, we now need to first multiply $\alpha\mathbf{X}^\top\mathbf{X}$ with $\boldsymbol{\theta}$ which is matrix multiplication of a $d \times d$ matrix with a $d \times 1$, which is $\mathcal{O}(d^2)$

For each of the t iterations, we now need to first multiply $\alpha \mathbf{X}^\top \mathbf{X}$ with $\boldsymbol{\theta}$ which is matrix multiplication of a $d \times d$ matrix with a $d \times 1$, which is $\mathcal{O}(d^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(d)$ for each iteration.

For each of the t iterations, we now need to first multiply $\alpha \mathbf{X}^\top \mathbf{X}$ with $\boldsymbol{\theta}$ which is matrix multiplication of a $d \times d$ matrix with a $d \times 1$, which is $\mathcal{O}(d^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(d)$ for each iteration.

What is overall computational complexity?

For each of the t iterations, we now need to first multiply $\alpha \mathbf{X}^\top \mathbf{X}$ with $\boldsymbol{\theta}$ which is matrix multiplication of a $d \times d$ matrix with a $d \times 1$, which is $\mathcal{O}(d^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(d)$ for each iteration.

What is overall computational complexity?

$\mathcal{O}(td^2) + \mathcal{O}(d^2 n) = \mathcal{O}((t+n)d^2)$

If we do not rewrite the expression $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$
For each iteration, we have:

- Computing $\mathbf{X}\boldsymbol{\theta}$ is $\mathcal{O}(nd)$
- Computing $\mathbf{X}\boldsymbol{\theta} - \mathbf{y}$ is $\mathcal{O}(n)$

If we do not rewrite the expression $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$
For each iteration, we have:

- Computing $\mathbf{X}\boldsymbol{\theta}$ is $\mathcal{O}(nd)$
- Computing $\mathbf{X}\boldsymbol{\theta} - \mathbf{y}$ is $\mathcal{O}(n)$
- Computing $\alpha \mathbf{X}^\top$ is $\mathcal{O}(nd)$
- Computing $\alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(nd)$
- Computing $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(n)$

If we do not rewrite the expression $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$
For each iteration, we have:

- Computing $\mathbf{X}\boldsymbol{\theta}$ is $\mathcal{O}(nd)$
- Computing $\mathbf{X}\boldsymbol{\theta} - \mathbf{y}$ is $\mathcal{O}(n)$
- Computing $\alpha \mathbf{X}^\top$ is $\mathcal{O}(nd)$
- Computing $\alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(nd)$
- Computing $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(n)$

What is overall computational complexity?

If we do not rewrite the expression $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$
For each iteration, we have:

- Computing $\mathbf{X}\boldsymbol{\theta}$ is $\mathcal{O}(nd)$
- Computing $\mathbf{X}\boldsymbol{\theta} - \mathbf{y}$ is $\mathcal{O}(n)$
- Computing $\alpha \mathbf{X}^\top$ is $\mathcal{O}(nd)$
- Computing $\alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(nd)$
- Computing $\boldsymbol{\theta} = \boldsymbol{\theta} - \alpha \mathbf{X}^\top (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$ is $\mathcal{O}(n)$

What is overall computational complexity?
$\mathcal{O}(ndt)$